AGID | Agenzia per l'Italia Digitale

Raccomandazioni in merito allo standard Transport Layer Security (TLS)

Versione 2.0 del 13/12/2022

Versione	Data	Tipologia modifica
1	21/05/2021	Prima emissione
2	13/12/2022	Allineamento ad avviso AgID nr. 18 del 15 aprile 2021

Sommario

Introduzione

Il presente Allegato delle Linee Guida Tecnologie e standard per la sicurezza dell'interoperabilità tramite API dei sistemi informatici individuano raccomandazioni in merito ai protocolli di sicurezza e alle Cipher Suite rappresentanti lo stato dell'arte al momento della sua stesura.

Capitolo 1

Ambito di applicazione

1.1 Soggetti destinatari

L'Allegato è destinato ai soggetti di cui al comma 2 dell'articolo 2 del CAD, che la attuano nella realizzazione dei propri sistemi informatici che fruiscono o erogano dati e/o servizi digitali di/ad altri soggetti tramite API.

L'Allegato è rivolto ai soggetti privati che devono interoperare con la Pubblica Amministrazione per fruire di dati e/o servizi tramite sistemi informatici tramite API.

Capitolo 2

Riferimenti e sigle

2.1 Note di lettura del documento

Conformemente alle norme ISO/IEC Directives, Part 3 per la stesura dei documenti tecnici la presente linea guida utilizzerà le parole chiave «DEVE», «DEVONO», «NON DEVE», «NON DEVONO», «DOVREBBE», «NON DOVREBBE», «PUO'», «POSSONO» e «OPZIONALE», la cui interpretazione è descritta di seguito.

- **DEVE** o **DEVONO**, indicano un requisito obbligatorio per rispettare la linea guida;
- **NON DEVE** o **NON DEVONO**, indicano un assoluto divieto delle specifiche;
- **DOVREBBE** o **NON DOVREBBE**, indicano che le implicazioni devono essere comprese e attentamente pesate prima di scegliere approcci alternativi;
- **PUÓ** o **POSSONO** o l'aggettivo **OPZIONALE**, indica che il lettore può scegliere di applicare o meno senza alcun tipo di implicazione la specifica.

2.2 Riferimenti Normativi

Sono riportati di seguito gli atti normativi di riferimento del presente documento.

[CAD]	decreto legislativo 7 marzo 2005, n. 82 recante «Codice dell'Amministrazione Digitale»
[EIF]	European Interoperability Framework (EIF)
[CE 2008/1205]	Regolamento (CE) n. 1205/2008 della Commissione del 3 dicembre 2008 recante attuazione della direttiva 2007/2/CE del Parlamento europeo e del Consiglio per quanto riguarda i metadati
[D.lgs. 196/2003]	Codice in materia di protezione dei dati personali
[UE 679/2016]	Regolamento (UE) 2016/679 del 27 aprile 2016 relativo alla protezione delle persone fisiche con riguardo al trattamento dei dati personali (in breve GDPR)
[UE 910/2014]	Regolamento (UE) n. 910/2014 del 23 luglio 2014 in materia di identificazione elettronica e servizi fiduciari per le transazioni elettroniche nel mercato interno (in breve eIDAS)

2.3 Termini e definizioni

Di seguito si riportano gli ACRONIMI che verranno utilizzati nella presente Linee Guida:

[AgID]	Agenzia per l'Italia Digitale
[AES]	Advanced Encryption Standard
[AEAD]	Authenticated Encryption with Associated Data
[DEA]	Data Encryption Algorithm
[DHE]	Diffie-Hellman in ephemeral mode
[DSA]	Digital Signature Algorithm
[ECC]	Elliptic Curve Cryptography
[ECDSA]	Elliptic Curve Digital Signature Algorithm
[HKDF]	Expand Key Derivation Function
[HMAC]	Keyed-hash Message Authentication Code
[IETF]	Internet Engineering Task Force
[KDF]	Key Derivation Function
[MD5]	Message Digest Algorithm
[PFS]	Perfect Forward Secrecy
[PSK]	Pre-Shared Key
[RFC]	Request For Comments
[RSA]	Rivest, Shamir e Adleman
[SHA]	Secure Hash Algorithm
[TDEA]	Triple Data Encryption Algorithm
[TLS]	Transport Layer Security
[3DES]	Triple Data Encryption Standard

Capitolo 3

Transport Layer Security

3.1 Versioni TLS disponibili

Ad oggi sono disponibili le seguenti versioni TLS:

- TLS 1.3 (pubblicato nel 2018)
- TLS 1.2 (pubblicato nel 2008)

TLS 1.0 e 1.1 sono protocolli obsoleti che non supportano i moderni algoritmi crittografici e risultano vulnerabili ad attacchi. Questi due protocolli sono da ritenersi deprecati come da comunicazioni di Google[1], Microsoft[2], Cisco[3], Apple[4] e Mozilla[5].

3.2 Suite di cifratura (cipher suite)

Il supporto crittografico in TLS è fornito attraverso l'uso di varie suite di crittografia. Una suite di cifratura definisce una combinazione di algoritmi per lo scambio di chiavi e per fornire riservatezza e integrità della sessione durante lo scambio di messaggi. Al momento della negoziazione di una sessione TLS il client presenta una serie di cipher suite supportate che propone al server il quale ne seleziona una.

[1] Cf. https://security.googleblog.com/2018/10/modernizing-transport-security.html
[2] Cf. https://docs.microsoft.com/en-us/office365/troubleshoot/o365-security/tls-1-2-in-office-365-gcc
[3] Cf. https://support.umbrella.com/hc/en-us/articles/360033350851-End-of-Life-for-TLS-1-0-1-1
[4] Cf. https://webkit.org/blog/8462/deprecation-of-legacy-tls-1-0-and-1-1-versions/
[5] Cf. https://blog.mozilla.org/security/2018/10/15/removing-old-versions-of-tls/

3.2.1 Suite di cifratura versione 1.2

Le suite di crittografia TLS 1.2 contengono quattro singoli algoritmi che rendono sicuro il canale in fase di handshake.

Una suite di cifratura può, ad esempio, essere composta dai seguenti 4 gruppi di algoritmi:

FUNZIONE	**ALGORITMO**
Scambio di chiavi	RSA, Diffie-Hellman, ECDH, SRP, PSK
Autenticazione	RSA, DSA, ECDSA
Cifratura dati	RC4, 3DES, AES
Hashing	HMAC-SHA256, HMAC-SHA1, HMAC-MD5

Un esempio di cipher suite TLS 1.2 con i relativi algoritmi utilizzati delle varie fasi è la seguente:

- **TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256**

3.2.2 Suite di cifratura versione 1.3

La nuova versione di TLS ha portato con sé molti miglioramenti. Tra i più importanti è stata la dismissione di algoritmi e cifrari vulnerabili o non adatti a garantire PFS, tra i quali:

- RC4 Stream Cipher
- RSA Key Exchange
- SHA-1 Hash Function
- CBC (Block) Mode Ciphers
- MD5 Algorithm
- Vari gruppi di Diffie-Hellman non-ephemeral
- EXPORT-strength ciphers
- DES
- 3DES

Inoltre, è stata migliorata la fase di handshake, dimezzando la latenza della crittografia e riducendo il tempo di handshake.

Le suite di crittografia TLS 1.3 non includono più gli algoritmi di scambio chiave e firma e l'autenticazione è stata unita alla crittografia in un unico algoritmo di tipo AEAD (Authenticated Encryption with Associated Data). Ciò ha semplificato le possibili combinazioni di cipher suite.

Attualmente sono solo cinque le cipher disponibili per TLS 1.3:

- **TLS_AES_256_GCM_SHA384**
- **TLS_CHACHA20_POLY1305_SHA256**
- **TLS_AES_128_GCM_SHA256**
- **TLS_AES_128_CCM_8_SHA256**
- **TLS_AES_128_CCM_SHA256**

COMPONENTE	**CONTENUTO**
TLS	La stringa «TLS»
AEAD	Algoritmo AEAD usato per la protezione dei dati
HASHdati	Algoritmo di hash usato con HKDF

Capitolo 4

Requisiti minimi lato server

Questa sezione fornisce una serie minima di requisiti che un server deve implementare per soddisfare queste linee guida, ma non ha l'obiettivo di fornire alcuna indicazione implementativa.

Si ribadisce che il presente documento verrà aggiornato all'eventuale insorgere di problemi di sicurezza legati ai protocolli e/o algoritmi.

4.1 Versioni TLS raccomandate

I servizi esposti DEVONO utilizzare la versione TLS 1.2, o superiori, e DOVREBBERO rifiutare la negoziazione di una versione inferiore.

Versioni precedenti del protocollo sono insicure o contengono vulnerabilità note.

Periodicamente bisogna controllare tutte le versioni e rimanere aggiornati per evitare configurazioni errate e nuove vulnerabilità.

4.2 Suite di cifratura raccomandate

Per agevolare la configurazione del protocollo TLS, Mozilla fornisce un tool[6] che genera configurazioni sicure dei suoi principali software.

DOVREBBE essere utilizzato nella configurazioni «Modern» (rivolte ai client con TLS 1.3 e che non necessitano di retro-compatibilità) e «Intermediate» (compatibili con la maggior parte dei client).

NON DOVREBBE essere utilizzato nella configurazioni di tipo «Old» perché potrebbero includere cipher suite vulnerabili.

[6] Cf. https://wiki.mozilla.org/Security/Server_Side_TLS

Di seguito sono elencate le versioni minime dei client per ciascuna categoria di configurazione:

	Modern	Intermediate	Old
FIREFOX	63	27	1
ANDROID	10.0	4.4.2	2.3
CHROME	70	31	1
EDGE	75	12	12
INTERNET EXPLORER		11 (Win7)	8 (WinXP)
JAVA	11	8u31	6
OPENSSL	1.1.1	1.0.1	0.9.8
OPERA	57	20	5
SAFARI	12.1	9	1

4.2.1 Cipher suite Modern

Sono accettate le suite di cifratura con le seguenti caratteristiche:

- Versione TLS: 1.3 (la 1.2 non è accettata)
- Tipo di certificato: ECDSA (P-256)
- Curva TLS: X25519, prime256v1, secp384r1
- Durata del certificato: 90 giorni

Suite di cifratura (TLS 1.3):

- **TLS_AES_128_GCM_SHA256**
- **TLS_AES_256_GCM_SHA384**
- **TLS_CHACHA20_POLY1305_SHA256**

Note: Le suite di cifratura moderne sono più sicure, ma potrebbero non essere compatibili con client obsoleti, rendendo inutilizzabile l'applicazione.

4.2.2 Cipher suite Intermediate

Sono accettate le suite di cifratura con le seguenti caratteristiche:

- Versione TLS: 1.3, 1.2
- Curva TLS: X25519, prime256v1, secp384r1

- Tipo di certificato: ECDSA (P-256) (raccomandato) o RSA (2048 bits)
- Durata del certificato: da 90 giorni (raccomandato) a 366 giorni
- Dimensione del parametro DH: 2048 (solo per Intermediate RFC7919)

Suite di cifratura (TLS 1.3):

- **TLS_AES_128_GCM_SHA256**
- **TLS_AES_256_GCM_SHA384**
- **TLS_CHACHA20_POLY1305_SHA256**

Suite di cifratura (TLS 1.2):

- **ECDHE-ECDSA-AES128-GCM-SHA256**
- **ECDHE-RSA-AES128-GCM-SHA256**
- **ECDHE-ECDSA-AES256-GCM-SHA384**
- **ECDHE-RSA-AES256-GCM-SHA384**
- **ECDHE-ECDSA-CHACHA20-POLY1305**
- **ECDHE-RSA-CHACHA20-POLY1305**
- **DHE-RSA-AES128-GCM-SHA256**
- **DHE-RSA-AES256-GCM-SHA384**

Capitolo 5

Ulteriori raccomandazioni

5.1 Rinegoziazione della sessione

La rinegoziazione di una sessione TLS è vulnerabile ad una serie di attacchi. Le implementazioni utilizzate DEVONO rispettare le indicazioni contenute in RFC5746[7]. Un server DOVREBBE rifiutare una rinegoziazione della sessione iniziata dal client.

Analogamente a quanto indicato in RFC5756[8], quando si utilizza HTTP/2 RFC7540[9] con TLS 1.3, il servizio NON DEVE permettere la post-handshake authentication, come indicato in RFC8740[10].

Considerazioni simili valgono in tutti i contesti in cui richieste multiple HTTP vengono trasmesse con meccanismi di multiplexing su singola connessione.

5.2 Compressione TLS

La compressione TLS DOVREBBE essere disabilitata; essa è stata rimossa dalla versione 1.3 di TLS perché sfruttata in passato da diversi exploit, tra cui il noto CRIME[11].

5.3 Estensione Heartbeat

L'estensione Heartbeat specificata in RFC 6520[12] permette di prolungare la durata di una connessione TLS senza dover eseguire una rinegoziazione della sessione. Questa estensione è stata utilizzata in Heartbleed, con cui l'attaccante è in grado di accedere ad alcune aree di memoria del server che potrebbero contenere dati riservati. L'uso dell'estensione Heartbeat è NON RACCOMANDATO e nel caso fosse necessario il suo utilizzo, si raccomanda di verificare che non sia vulnerabile a Heartbleed.

[7] Cf. https://tools.ietf.org/html/rfc5746
[8] Cf. https://tools.ietf.org/html/rfc5756
[9] Cf. https://tools.ietf.org/html/rfc7540
[10] Cf. https://tools.ietf.org/html/rfc8740
[11] Cf. https://en.wikipedia.org/wiki/CRIME
[12] Cf. https://tools.ietf.org/html/rfc6520

Capitolo 6

TLS 1.2 vs TLS 1.3 versioni a confronto

Il protocollo TLS 1.3 è stato definito in RFC 8446[13] nell'agosto 2018 ed è più sicuro e veloce rispetto a TLS 1.2 RFC 5246[14]. Le principali differenze includono:

- L'elenco degli algoritmi simmetrici supportati è stato epurato da tutti gli algoritmi legacy. Gli algoritmi rimanenti utilizzano algoritmi di crittografia autenticata con dati associati (AEAD) come ad esempio ChaCha20, Poly1305, Ed25519, x25519 e x448.
- È stata aggiunta una modalità zero-RTT (0-RTT) che elimina un round-trip durante la fase di configurazione della connessione.
- Le suite di cifratura statiche RSA e Diffie-Hellman sono state rimosse.
- Tutti i messaggi di handshake dopo "ServerHello" sono ora cifrati.
- Le funzioni di derivazione delle chiavi sono state ri-progettate, con la funzione di derivazione delle chiavi di estrazione ed espansione basata su HMAC (HKDF) utilizzata come primitiva.
- Gli stati dell'handshake sono stati ristrutturati per essere più coerenti e sono stati rimossi i messaggi superflui.
- ECC è ora nelle specifiche di base del TLS 1.3 e include nuovi algoritmi di firma.
- Sono stati inoltre apportati altri miglioramenti crittografici, tra cui:
    - la modifica del Padding RSA al fine di utilizzare lo schema RSA Probabilistic Signature Scheme (RSASSA-PSS) definito da RFC 8017;
    - la rimozione della compressione, dell'algoritmo di firma digitale (DSA) e dei gruppi Ephemeral Diffie Hellman (DHE) personalizzati.
- la modifica del Padding RSA al fine di utilizzare lo schema RSA Probabilistic Signature Scheme (RSASSA-PSS) definito da RFC 8017;

[13] Cf. https://tools.ietf.org/html/rfc8446
[14] Cf. https://tools.ietf.org/html/rfc5246

- la rimozione della compressione, dell'algoritmo di firma digitale (DSA) e dei gruppi Ephemeral Diffie Hellman (DHE) personalizzati.

Come rappresentato nella seguente figura, per il TLS 1.2 sono necessari due round trip per completare l'handshake del TLS. La versione TLS 1.3 richiede un solo round-trip, che a sua volta diminuisce la latenza della crittografia. Si ottiene quindi un risparmio medio di tempo sull'handshake consentendo connessioni cifrate più veloci.

Inoltre, i meccanismi come TLS False Start e Zero Round Trip Time (0-RTT) del TLS 1.3 consentono di ridurre ulteriormente il tempo richiesto per l'handshake con gli host ai quali il client si è già collegato in precedenti sessioni.

Infine, TLS 1.3 offre una maggiore protezione contro i cosiddetti "downgrade attacks" ovvero i tentativi posti in essere da parte di un attaccante per indurre il server all'utilizzo di una vecchia versione del protocollo TLS soggetta a vulnerabilità note.